1. ChangeDetectionStrategy
2. Angular 2 security
3. AOT compiler
4. Different types of metadata – host
5. Pagination
6. <https://stackoverflow.com/questions/38236313/angular2-inject-external-component-into-other-component-via-directive-not-worki>
7. How to inject another module component into another module’s component?

Try to export it

@NgModule({

declarations: [TaskCardComponent],

imports: [MdCardModule],

exports: [TaskCardComponent] <== this line

})

export class TaskModule{}

<https://stackoverflow.com/questions/39601784/angular-2-use-component-from-another-module>

1. How to use one’s component properties into another component
2. Metadata tells Angular how to process a class. It's not a component until you tell Angular about it. To tell Angular that HeroListComponent is a component, attach **metadata** to the class. In TypeScript, you attach metadata by using a **decorator**.
3. In two-way binding, a data property value flows to the input box from the component as with property binding. The user's changes also flow back to the component.
4. Apply conditional class

<div [class.active]=”row === 2”></div>

**Structural** directives alter layout by adding, removing, and replacing elements in DOM. (\*ngFor, \*ngIf)

1. The backtick (`)—which is not the same character as a single quote (')—allows you to compose a string over several lines.
2. **Attribute** directives alter the appearance or behavior of an existing element. (\*ngModel)
3. With \*ngIf, Angular isn't showing and hiding the message. It is adding and removing the paragraph element from the DOM.
4. Template reference variable